Name: Wei Wang
Date: 8/28/2022
Course: IT FDN 110 B Su 22: Foundations Of Programming: Python

GitHub repository:

https://github.com/vivian39/IntroToProg-Python-Mod07

GitHub Webpage:

https://github.com/vivian39/IntroToProg-Python-Mod07#introtoprog-python-mod07

Assignment07: Files & Exceptions

Introduction

This module introduces exception handling and Python’s pickling module. I will write a Python Script that allows a user to input a list of IDs and names and then pickle the list and save it to a .dat file.

Research Exception Handling in Python

I read the article "Python Exception handling" from geeksforgeeks.org (https://www.geeksforgeeks.org/python-exception-handling/).

It's a good introduction of "Exception handling" Firstly because it's newly published so the information should be up to date; secondly, the language is very concise and well structured so the beginner Python user can catch the concept really easily and fast. Lastly, it provided examples and pictures to help explain each concept. Additionally, there is a YouTube video attached at the end so the audience can easily learn it by watching the video. Learning in different ways, from different resources really help the users have a better understanding of what they are learning.

Research Pickling in Python

"Python pickle module is used for serializing and de-serializing python object structures. The process to converts any kind of python objects (list, dict, etc.) into byte streams (0s and 1s) is called pickling or serialization or flattening or marshalling. We can converts the byte stream (generated through pickling) back into python objects by a process called as unpickling." I learned this concept from an article named "Python Pickling"(https://www.tutorialspoint.com/python-pickling)

It also introduced an example of pickling a simple list and an example of unpickling a simple list; pickle exceptions and the pros and cons of pickling.

Create a Python Script(in a Mac book)

Assignment requirements: create a new script that demonstrates how Pickling and Structured error handling work.

I created a new sub-folder called Assignment07 inside the _PythonClass folder; created a new project in PyCharm that uses the _PythonClass\Assignment07 folder as its location; created a file called, "Assigment07.py," in my project; added code to the script that performs the assignment’s task.

Part1: # Declare variables and constants
strFileName = 'AppData.dat'
lstCustomer = [ ]

Part2: # Processing data

define a function to save data to a file
define a function to read data from a file
define a function to input data to a list

Part3: # Presentation
get ID and NAME From user, then store it in a list object
store the list object into a binary file
read the data from the file into a new list object and display the contents

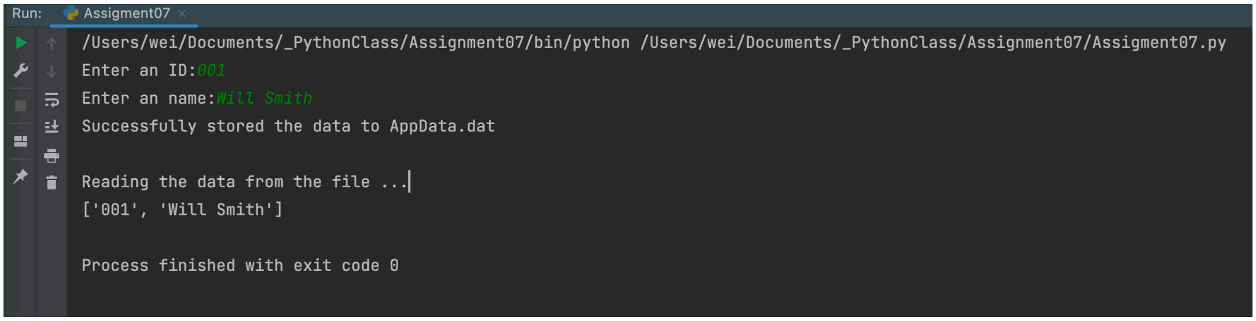
Code:

```
# -------------------------------------------------- #
# Title: Lab7-1
# Description: A simple example of storing data in a binary file
# ChangeLog: (Who, When, What)
# <WeiWang>,<8.28.2022>,Created Script
# ------------------------------------------------ #

import pickle  # This imports code from another code file!

# Data -------------------------------------------- #
strFileName = 'AppData.dat'
lstCustomer = []


# Processing ------------------------------------- #
def save_data_to_file(file_name, list_of_data):
    # Use binary mode to save data
    with (open(file_name, "wb")) as openfile:
        pickle.dump(list_of_data, openfile)


def read_data_from_file(file_name):
    # Use binary mode to read data
    with (open(file_name, "rb")) as openfile:
        return pickle.load(openfile)


def input_data_to_list():
    id = input("Enter an ID:")
    name = input("Enter an name:")
    return [id, name]


# Presentation ------------------------------------ #
# TODO: Get ID and NAME From user, then store it in a list object
lstCustomer = input_data_to_list()
# TODO: store the list object into a binary file
save_data_to_file(strFileName, lstCustomer)
print("Successfully stored the data to " + strFileName)

# TODO: Read the data from the file into a new list object and display the
contents
print("\nReading the data from the file ...")
data_from_file = read_data_from_file(strFileName)
print(data_from_file)
```

Run the code

Run the script both in PyCharm and an OS command/shell window and capture images of it working on my computer

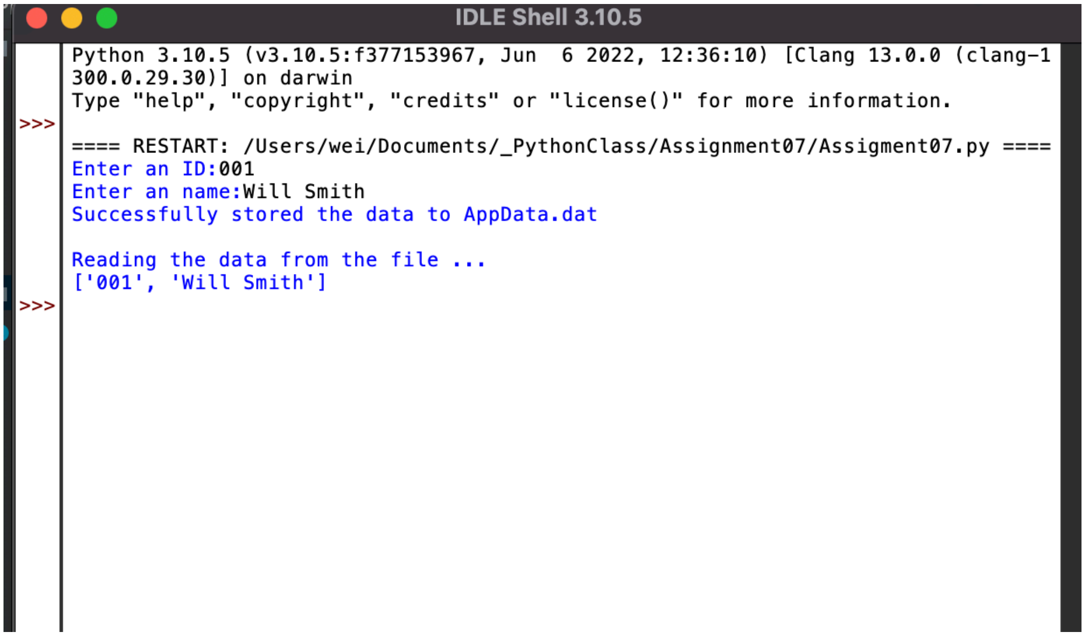
This is the screen shot of the script running in PyCharm(Figure 1)



```
/Users/wei/Documents/_PythonClass/Assignment07/bin/python /Users/wei/Documents/_PythonClass/Assignment07/Assigment07.py
Enter an ID:001
Enter an name:Will Smith
Successfully stored the data to AppData.dat

Reading the data from the file ...
['001', 'Will Smith']

Process finished with exit code 0
```

Figure 1: the screen shot of the script running in PyCharm

This is the screen shot of the script running in IDLE(Figure 2)

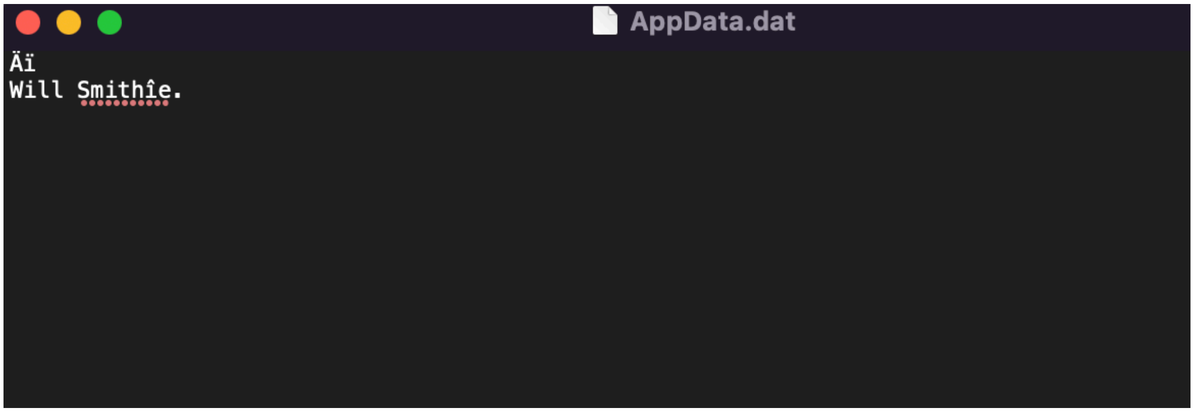


```
    Python 3.10.5 (v3.10.5:f377153967, Jun  6 2022, 12:36:10) [Clang 13.0.0 (clang-1
    300.0.29.30)] on darwin
    Type "help", "copyright", "credits" or "license()" for more information.
>>>
    ==== RESTART: /Users/wei/Documents/_PythonClass/Assignment07/Assigment07.py ====
    Enter an ID:001
    Enter an name:Will Smith
    Successfully stored the data to AppData.dat

    Reading the data from the file ...
    ['001', 'Will Smith']
>>>
```

Figure 2: the screen shot of the script running in IDLE

Verify that it Worked

Locate the binary file and open it in a text editor(Figure 3).



```
Äï
Will Smithîe.
```

Figure 3: Verifying that the file has data

Summary

In this module, I got to learn exception handling and Python’s pickling. I benefit from doing my own research and have a better understanding of the assignment.